# Modelling of Complex Systems IDP (Invariants) Project

senne.berden@kuleuven.be dorde.markovic@kuleuven.be

March 2022

## 1 Introduction

This part of the project focuses on proving a couple of invariants about the elevator dynamic system. The elevator was the first part of the project and hence you should do this part as a continuation of the previous part.

Expressing invariants should not take much time (it can be done in 15-30 minutes) but note that if your elevator theory is not correct and invariants are not holding, this can lead to the major revision of your solution. You can use this part of the project as a verification of your elevator specification.

## 2 Invariants

Your goal is to prove the following invariants (for more details on invariants check the exercise session 5):

1. Whenever the elevator is moving the elevator doors are closed.
2. The elevator doors are never open if there is no call waiting on the floor of the elevator.

## 3 Instructions and additional materials

To specify invariants create a new theory per invariant.

```
// If elevator moves the doors are closed!
theory move_inv : V_student {

}

//Doors are never open if there is no call waiting on that floor!
theory door_inv : V_student {

}
```

Verify these invariants using the following structure:

```
Structure S_Inv:V_student {
    Time = {0..20}
    Floor = {0..5}
    Start = 0
}
```

Use the following procedure to check invariants. Note that you will have to change your main method and call the provided method.

```
procedure proveInvariants(){
    print(isinvariant(T_student , move_inv , S_Inv))
    print(isinvariant(T_student , door_inv , S_Inv))
}
```

## 4 Output

The output of this part of the project is merged with the output of the first part of the project. So, you should not submit two but only one solution. You should add the code provided above (also available in a separate file on Toledo assignment for an easy copy) to your `elevator.idp` file and specify invariants.

Additionally, you should add one more field to your `report.txt` file, namely "Major revision:" and answer `yes` if this part of the project caused you to rewrite your elevator solution and `no` otherwise.

## 5 Theory restrictions

In order to use `isinvariant` method you have to make sure that your theory satisfies certain syntactic constraints:

- Your theory has to use `Time` as the time type. Time points are never used directly in your theory. Use constant `Start`, function `Next` and quantify over time. You should not use 0, $t + 1$, or $S(t)$.
- There are no restrictions on non-temporal (that do not contain terms designating time at all) formulas and rules.
- Use only single-state or bistate formulas in your theory.
- Definitions of temporal predicates (fluents, causality predicates, action predicates) consist of single-state rules of the form:

  $$\forall t \ldots (P(\ldots, t) \leftarrow \varphi[t])$$

  where $\varphi[t]$ is a single state formula, and bistate rules of the form:

  $$\forall t \ldots (P(\ldots, Next(t)) \leftarrow \varphi[t])$$

  where $\varphi[t]$ is a bistate formula.
- **Note** that if you follow these constraints you can not *define* the current state in terms of future time points.

Good luck!
*In case you have any questions, do not hesitate to contact the teaching assistants.*